Example for my workflow with `knitr::spin()`

Kirill Müller

October 24, 2013

Abstract

Markdown + LaTeX + `knitr::spin()` = fun.

Introduction

I love LaTeX. But sometimes it's just too verbose:

- To add an enumeration, I have to type `\begin{enumerate}`, `\end{enumerate}`, *and* an additional `\item` for each item
- Emphasis, teletype formatting, ... – all doable, but just a bit too cumbersome for everyday use

Of course there are nice editors with autocompletion and whatnot. But, for simple things, indeed I do prefer Markdown. Plus, RStudio is not yet a full-fledged LaTeX editor.

On the other hand, formulae are really nice in LaTeX: $2 \cdot 3 = 4$, $\sum_{i=1}^{k} m_i$, stuff like that. Cross-references, citations, ... To sum up, I want both.

Also, when doing a statistical analysis, I'm not going to ever copy-paste a number or a figure. I want everything to be computed by my R script and neatly knitted with `knitr`: $2 \cdot 3 = 6$, and this can be even checked:

```
stopifnot(2 * 3 == 4)
```

```
## Error: 2 * 3 == 4 is not TRUE
```

However, I don't want to *depend* on it: A simple `Rscript example.R` should run the code. That's where `spin()` comes to play: It will interpret `roxygen`-style comments as text, and everything else as chunks of R code (with an optional header that sets chunk options). This means that text and chunk options are seen as comments by the R engine, but interpreted sensibly by `spin()`: I'm getting the full power of `knitr`

with the option to run everything as a simple R script (without having to `tangle()` it first).

To combine this with the two other requirements above, I use a `Makefile` and a helper script that calls `knitr::spin()` to create Markdown + LaTeX, `pandoc` to convert this to plain LaTeX, and then `latexmk` to create a PDF.